Preparing Students for AI-Assisted Research Workflows

Authors

Alejandra Castillo <amcw2015@pomona.edu>, Department of Mathematics and Statistics, Pomona College, 0000-0002-4835-2190

Keywords

Education, Statistical Computing, AI tools

Abstract

When students can obtain a script that runs, it can be easy for them to believe they understand the programming behind it. AI tools make this tension more visible. Students can now generate code, examples, explanations, and error-message fixes quickly, even while they are still developing basic programming fluency. This can lower the barrier to entry for students who are new to computing, but it can also hide important gaps in understanding. Code may run without answering the right question, students may accept unfamiliar solutions they cannot explain, and statistical assumptions may be buried behind polished output.

In this talk, I share lessons from teaching undergraduate statistical computing, with examples from courses using R and Python for data analysis and modeling. I focus on a shift from teaching syntax as the central challenge to teaching computational judgment: reading code, debugging, checking whether output makes sense for the data and scientific context, documenting choices, and explaining results. I also discuss course redesigns that respond to AI-assisted coding, including giving students incorrect AI-generated output to critique and debug, adding written explanations of code and results, and using assessments that ask students to reason about data, models, and reproducibility. This talk connects these classroom experiences to broader questions about how educators can prepare future research software contributors to work responsibly in AI-assisted research workflows.

Connection to Mission, Goals, & Interests of US-RSE Community

As AI tools become more common in scientific workflows, the next generation of RSEs and research software contributors will need more than the ability to generate code. They will need to read, debug, explain, and maintain code in ways that support reliable research. They will also need to understand when a computational result is scientifically or statistically questionable, even if the code appears to run successfully.

This talk is intended for educators, mentors, and RSEs who support students or early-career researchers. It contributes to the US-RSE goals of resource dissemination and diversity by sharing teaching experiences that can help students with varied levels of prior computing experience participate more fully in research computing. By focusing on explanation, reproducibility, and judgment, the talk highlights skills that are central to sustainable research software and increasingly important in AI-assisted research environments.